

What is claimed is.

1 1. A system for verification of a system design,
2 comprising:

3 a test program generator that accepts a sequence of
4 statements including at least one event;

5 an event handling facility in said test program
6 generator; and

7 wherein responsive to a triggering condition of said
8 event said test program generator emits test program
9 instructions in response to one of a primary input stream
10 and an alternate input stream, said alternate input
11 stream being represented in a body of said event.

1 2. The system according to claim 1, wherein said
2 event comprises a plurality of events that are processed
3 in order of priority values thereof in said event
4 handling facility.

1 3. The system according to claim 1, wherein a condi-
2 tional statement of said event references a current state
3 of a test program that is generated by said test program
4 generator.

1 4. A method of test program generation, comprising
2 the steps of:

3 defining a set of statements, said set of statements
4 including an event;

5 responsive to said set of statements generating a
6 sequence of test program instructions for a target;

7 while performing said step of generating said
8 sequence of test program instructions determining if a
9 condition of said event is satisfied; and

10 responsive to said step of determining generating an
11 alternate sequence of test program instructions.

1 5. The method according to claim 4, wherein said step
2 of determining is performed by evaluating a state of said
3 target prior to inclusion of an instruction in said
4 sequence of test program instructions.

1 6. The method according to claim 4, wherein at least
2 a portion of said sequence of test program instructions
3 are randomly generated.

1 7. The method according to claim 4, wherein said
2 event comprises a plurality of events, each of said
3 events having a priority value, and said step of
4 determining if said condition is satisfied is performed
5 with respect to each of said events in an order of said
6 priority value thereof.

1 8. The method according to claim 4, wherein said
2 event has an identifying name attribute.

1 9. The method according to claim 4, wherein said
2 event has a triggering condition attribute.

1 10. The method according to claim 4, wherein said
2 event comprises an input stream entity.

1 11. The method according to claim 4, wherein said
2 event comprises an identifying name attribute, a
3 triggering condition attribute, a priority value and an
4 input stream entity.

1 12. A method of test program generation, comprising
2 the steps of:

3 providing a test program generation engine;

4 coupling said test program generation engine to a
5 design specification of a target, wherein said design
6 specification comprises a knowledge base;

7 coupling said test program generation engine to an
8 architectural simulator of said target;

9 introducing a set of statements into said test
10 program generation engine, said set of statements
11 including an event;

12 determining whether a triggering condition of said
13 event is satisfied;

14 in a first case, wherein said triggering condition is
15 not satisfied, causing said test program generation
16 engine to respond to said set of statements to generate a

17 first sequence of test program instructions that can be
18 executed on said target; and

19 in a second case, wherein said triggering condition
20 is satisfied, causing said test program generation engine
21 to respond to an alternate set of statements of said
22 event to generate a second sequence of test program
23 instructions that can be executed on said target.

1 13. The method according to claim 12, wherein at
2 least a portion of said first sequence of test program
3 instructions and said second sequence of test program
4 instructions is generated randomly.

1 14. The method according to claim 12, wherein said
2 step of determining is performed by evaluating a state of
3 said target prior to inclusion of an instruction in said
4 first sequence of test program instructions.

1 15. The method according to claim 12, wherein said
2 event comprises a plurality of events, each of said
3 events having a priority value, and said step of
4 determining is performed with respect to each of said
5 events in an order of said priority value thereof.

1 16. The method according to claim 12, wherein said
2 set of statements is introduced into said test program
3 generation engine as an input file.

1 17. The method according to claim 12, wherein said
2 event has an identifying name attribute.

1 18. The method according to claim 12, wherein said
2 event has a triggering condition attribute.

1 19. The method according to claim 12, wherein said
2 event comprises an input stream.

1 20. The method according to claim 12, wherein said
2 event comprises an identifying name attribute, a
3 triggering condition attribute, a priority value and an
4 input stream.

1 21. A computer software product, comprising a
2 computer-readable medium in which computer program
3 instructions are stored, which instructions, when read by
4 a computer, cause the computer to generate test programs
5 by performing the steps of:

6 accepting a set of statements, said set of statements
7 including an event;

8 responsive to said set of statements generating a
9 sequence of test program instructions for a target;

10 while performing said step of generating said
11 sequence of test program instructions determining if a
12 condition of said event is satisfied; and

13 responsive to said step of determining generating an
14 alternate sequence of test program instructions.

1 22. The computer software product according to
2 claim 21, further comprising the steps of accessing a
3 knowledge base having information of said target stored
4 therein, and said step of generating said sequence of
5 test program instructions comprises selecting members of
6 said sequence of test program instructions in accordance
7 with said information in said knowledge base, wherein
8 said step of selecting members is biased by said set of
9 statements.

1 23. The computer software product according to
2 claim 21, wherein at least a portion of said sequence of
3 test program instructions are randomly generated.

1 24. The computer software product according to
2 claim 21, wherein said event comprises a plurality of
3 events, each of said events having a priority value, and
4 said step of determining if said condition is satisfied
5 is performed with respect to each of said events in an
6 order of said priority value thereof.

1 25. The computer software product according to
2 claim 21, wherein said event has an identifying name
3 attribute.

1 26. The computer software product according to
2 claim 21, wherein said event has a triggering condition
3 attribute.

1 27. The computer software product according to
2 claim 21, wherein said event comprises a body that is a
3 template for generation of said alternate sequence of
4 test program instructions.

1 28. The computer software product according to
2 claim 21, wherein said event comprises an identifying
3 name attribute, a triggering condition attribute, a
4 priority value and a body.

1 29. A computer software product, comprising a
2 computer-readable medium in which computer program
3 instructions are stored, which instructions, when read by
4 a computer, cause the computer to generate test programs
5 by performing the steps of:

6 defining a test program generation engine in a
7 memory;

8 defining a design specification of a target in said
9 memory, wherein said design specification comprises a
10 knowledge base;

11 defining an architectural simulator of said target in
12 said memory;

13 coupling said test program generation engine to said
14 design specification;

15 coupling said test program generation engine to said
16 architectural simulator;

17 accepting a set of statements into said test program
18 generation engine, said set of statements including an
19 event;

20 responsive to said set of statements, and to
21 information in said knowledge base, causing said test
22 program generation engine to generate a test program
23 instruction that can be executed on said target;

24 thereafter determining in said architectural
25 simulator whether a triggering condition of said event is
26 satisfied by a simulated execution of said test program
27 instruction;

28 in a first case, wherein said triggering condition is
29 not satisfied, causing said test program generation
30 engine to respond to said set of statements to generate a
31 first sequence of test program instructions that can be
32 executed on said target; and

33 in a second case, wherein said triggering condition
34 is satisfied, causing said test program generation engine
35 to respond to an alternate set of statements of said
36 event to generate a second sequence of test program
37 instructions that can be executed on said target.

1 30. The computer software product according to
2 claim 29, wherein at least a portion of said first
3 sequence of test program instructions and said second
4 sequence of test program instructions is generated
5 randomly.

1 31. The computer software product according to
2 claim 29, wherein said step of determining is performed
3 by evaluating a state of said target prior to inclusion
4 of said test program instruction in one of said first
5 sequence of test program instructions and said second
6 sequence of test program instructions.

1 32. The computer software product according to
2 claim 31, wherein said step of evaluating said state is
3 performed prior to said simulated execution of said test
4 program instruction.

1 33. The computer software product according to
2 claim 31, wherein said step of evaluating said state is
3 performed subsequent to said simulated execution of said
4 test program instruction.

1 34. The computer software product according to
2 claim 31, wherein said step of evaluating said state is
3 performed a first time prior to a simulated execution of
4 said test program instruction and is performed a second
5 time subsequent to said simulated execution thereof.

1 35. The computer software product according to
2 claim 29, wherein said event comprises a plurality of
3 events, each of said events having a priority value, and
4 said step of determining is performed with respect to
5 each of said events in an order of said priority value
6 thereof.

1 36. The computer software product according to
2 claim 29, wherein said set of statements is introduced
3 into said test program generation engine as an input
4 file.

1 37. The computer software product according to
2 claim 29, wherein said event has an identifying name
3 attribute.

1 38. The computer software product according to
2 claim 29, wherein said event has a triggering condition
3 attribute.

1 39. The computer software product according to
2 claim 29, wherein said event comprises an input stream.

1 40. The computer software product according to
2 claim 29, wherein said event comprises an identifying
3 name attribute, a triggering condition attribute, a
4 priority value and an input stream.

1 41. A test program generator, comprising:
2 a test program generation engine;
3 a design specification of a target, wherein said
4 design specification comprises a knowledge base, wherein
5 said test program generation engine is coupled to said
6 design specification;
7 an architectural simulator of said target coupled to
8 said test program generation engine;
9 wherein said test program generation engine is
10 adapted to accept a set of statements, said set of
11 statements including an event;
12 responsive to said set of statements, and to
13 information in said knowledge base, said test program
14 generation engine generates a test program instruction
15 that can be executed on said target;
16 wherein said test program generation engine
17 determines whether a triggering condition of said event
18 is satisfied by a simulated execution of said test
19 program instruction;
20 in a first case, wherein said triggering condition is
21 not satisfied, responsive to said set of statements said
22 test program generation engine generates a first sequence
23 of test program instructions that can be executed on said
24 target; and
25 in a second case, wherein said triggering condition
26 is satisfied, responsive to an alternate input stream in
27 said event said test program generation engine, generates

28 a second sequence of test program instructions that can
29 be executed on said target.

1 42. The test program generator according to claim 41,
2 further comprising a design simulator for simulating said
3 simulated execution.

1 43. The test program generator according to claim 41,
2 wherein at least a portion of said first sequence of test
3 program instructions and said second sequence of test
4 program instructions is generated randomly.

1 44. The test program generator according to claim 41,
2 wherein said architectural simulator determines whether
3 said triggering condition of said event is satisfied by
4 evaluating a state of said target prior to inclusion of
5 said test program instruction in one of said first
6 sequence of test program instructions and said second
7 sequence of test program instructions.

1 45. The test program generator according to claim 44,
2 wherein said step of evaluating said state is performed
3 prior to said simulated execution of said test program
4 instruction.

1 46. The test program generator according to claim 44,
2 wherein said step of evaluating said state is performed
3 subsequent to said simulated execution of said test
4 program instruction.

1 47. The test program generator according to claim 44,
2 wherein said step of evaluating said state is performed a
3 first time prior to said simulated execution of said test
4 program instruction and is performed a second time
5 subsequent to said simulated execution.

1 48. The test program generator according to claim 41,
2 wherein said event comprises a plurality of events, each
3 of said events having a priority value, and said step of
4 determining is performed with respect to each of said
5 events in an order of said priority value thereof.

1 49. The test program generator according to claim 41,
2 wherein said set of statements is introduced into said
3 test program generation engine as an input file.

1 50. The test program generator according to claim 41,
2 wherein said event has an identifying name attribute.

1 51. The test program generator according to claim 41,
2 wherein said event has a triggering condition attribute.

41

1 52. The test program generator according to claim 41,
2 wherein said event comprises an input stream entity.

1 53. The test program generator according to claim 41,
2 wherein said event comprises an identifying name
3 attribute, a triggering condition attribute, a priority
4 value and an input stream entity.